

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

⦿ BLACK BORDERS

- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS

⦿ BLACK OR VERY BLACK AND WHITE DARK PHOTOS

- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.**

(12) UK Patent Application (19) GB (11) 2 296 798 (13) A

(43) Date of A Publication 10.07.1996

(21) Application No 9500173.1

(22) Date of Filing 05.01.1995

(71) Applicant(s)

Spring Consultants Limited

(Incorporated in the United Kingdom)

Unit 5, Ashbrook Mews, Westbrook Street,
BLEWBURY, Oxon, OX11 9QA, United Kingdom

(72) Inventor(s)

Andrew Paul George Randall
Norman Hamilton Burkie

(74) Agent and/or Address for Service

Atkinson & Co
Sixth Floor, High Holborn House, 52-54 High Holborn,
LONDON, WC1V 6SE, United Kingdom

(51) INT CL⁶
G06F 12/02

(52) UK CL (Edition O)
G4A AMX

(56) Documents Cited
None

(58) Field of Search
UK CL (Edition N) G4A AMX
INT CL⁶ G06F 12/02
ONLINE DATABASES : WPI, INSPEC

(54) Storing data efficiently on a RAID

(57) Data is stored in such a way that a plurality of user terminals 16 are given access to a large storage volume in the form of a redundant array of inexpensive drives (RAID 5) 21 to 25. The large storage volume is divided into a plurality of storage blocks and each of said blocks has a capacity which is smaller than the size of an emulated logical disc drive. In operation, physical blocks of data are mapped onto an emulated drive as storage is required up to a predetermined capacity.

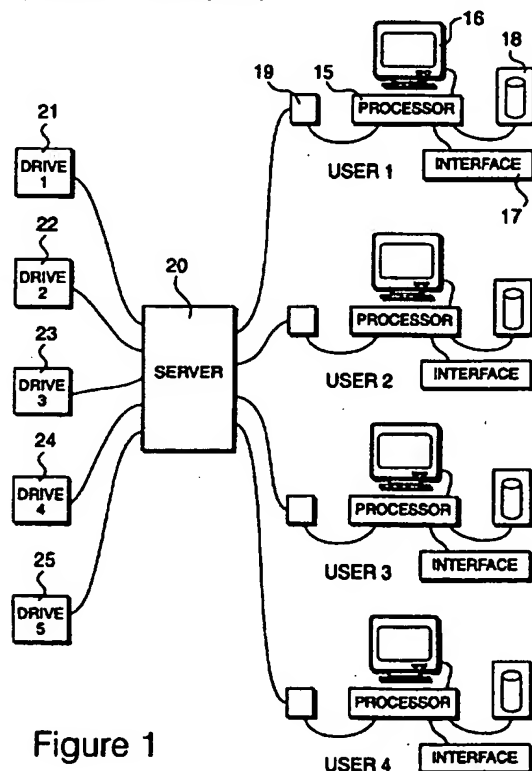


Figure 1

GB 2 296 798 A

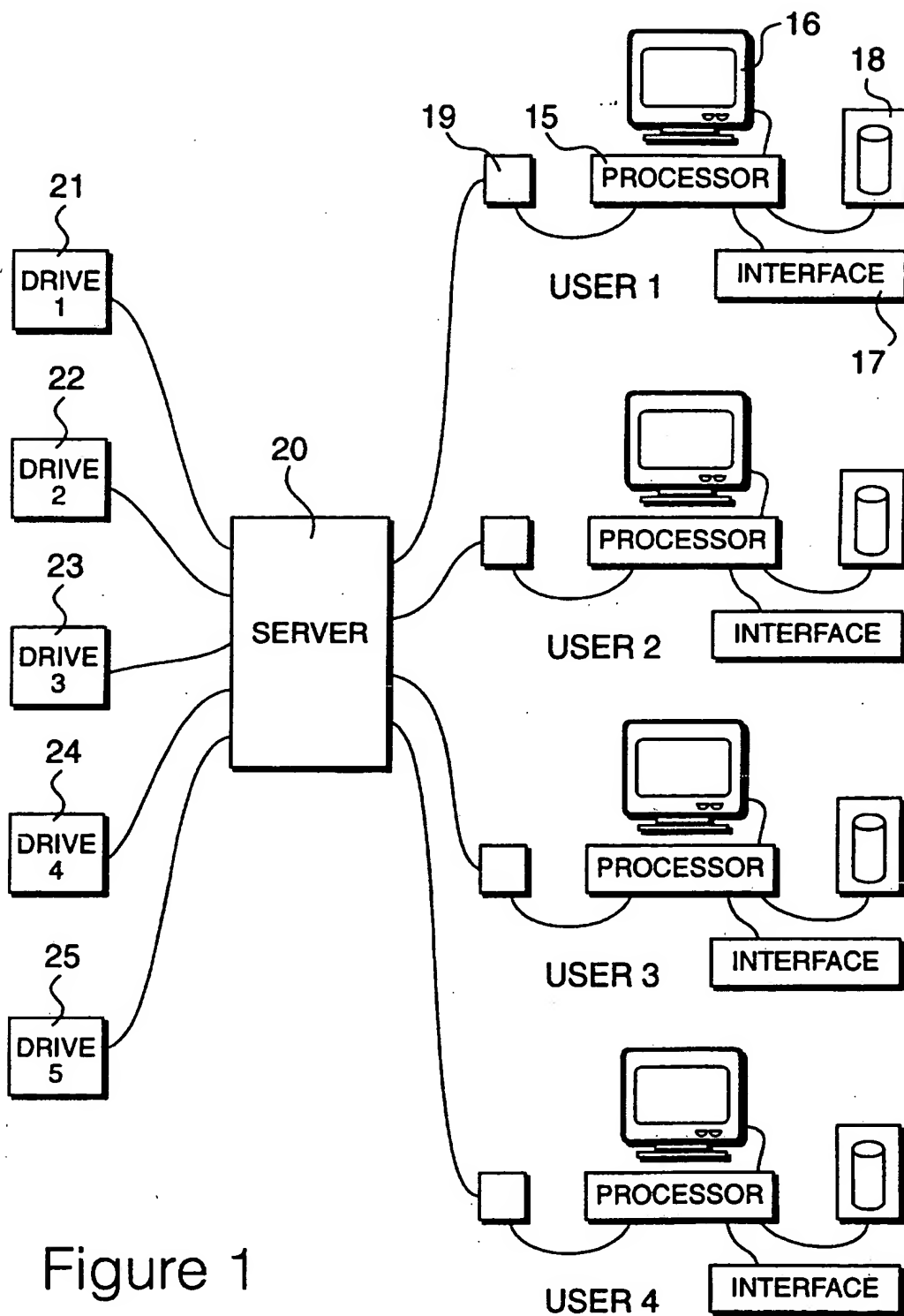


Figure 1

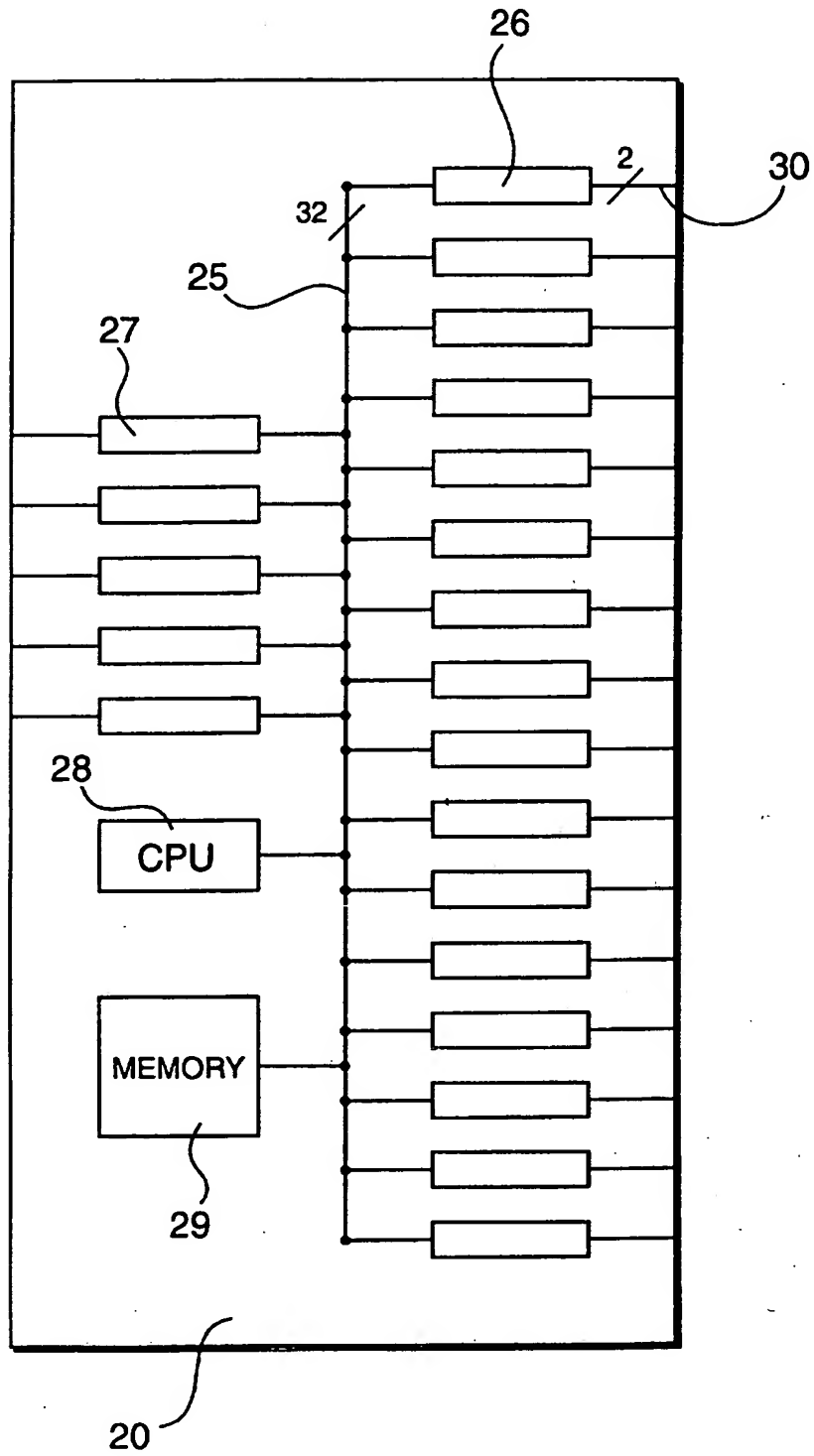


Figure 2

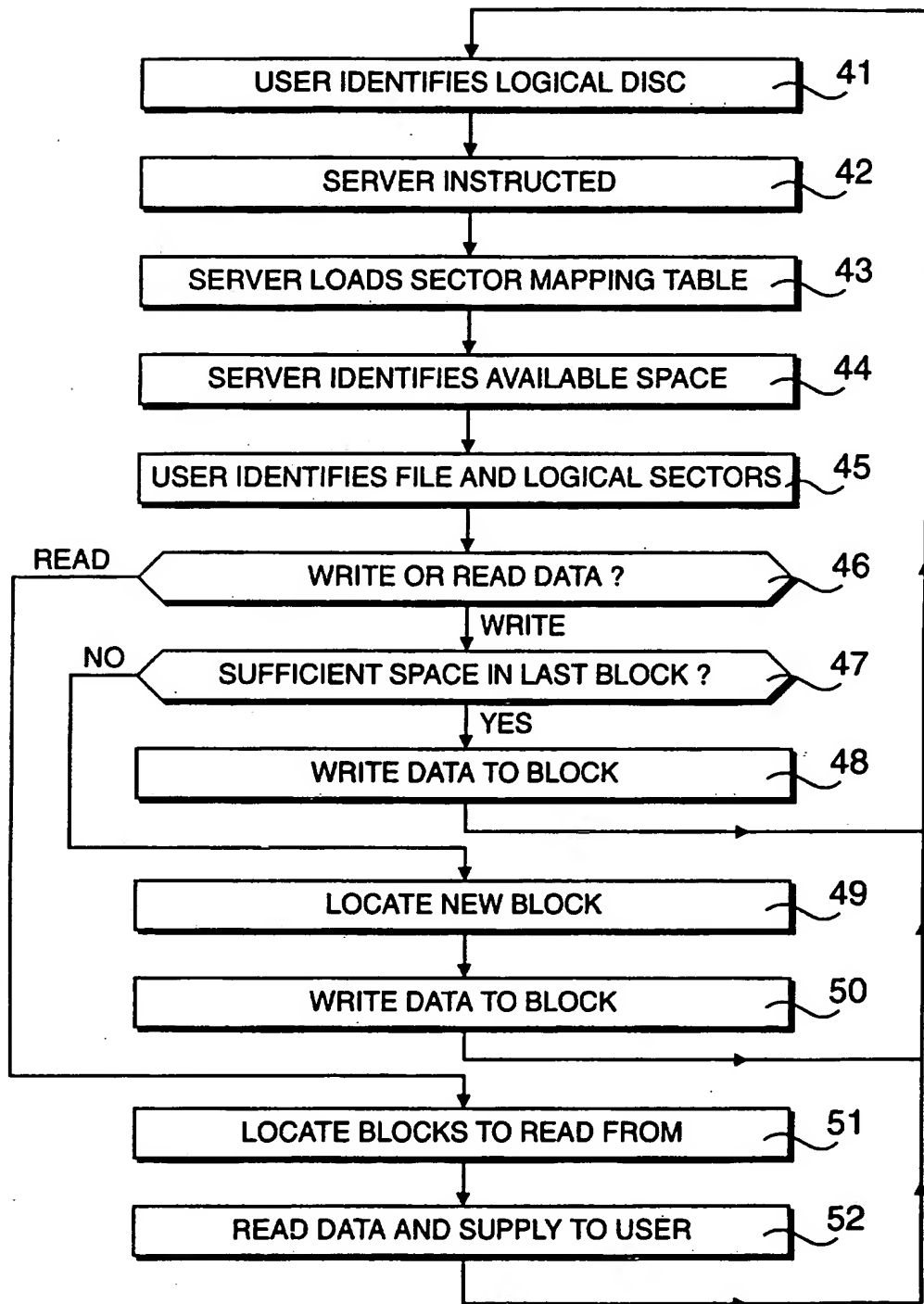


Figure 3

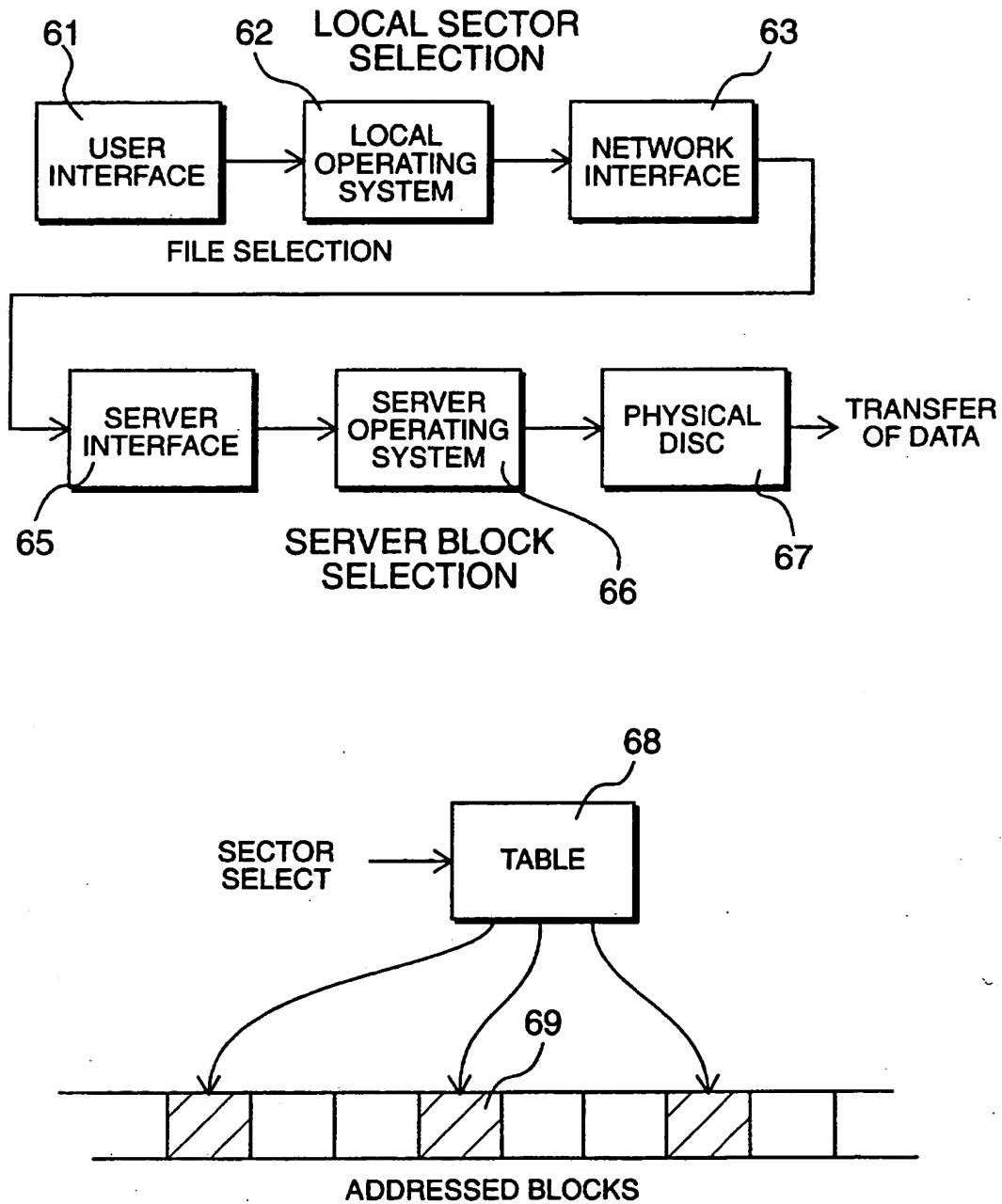


Figure 4

STORING DATA

The present invention relates to storing data. In particular, the present invention relates to an environment in which a plurality of user terminals
5 have shared access to a large storage volume.

Systems are known in which data storing devices, often referred to as volumes, are shared amongst a plurality of user terminals or workstations. Typically, the volume is associated with a local workstation, referred to as a server, and the totality of the workstations are interconnected by a network,
10 such as an ethernet. Such an arrangement provides efficient shared access to files provided that the amount of data contained within each file is small compared to the transmission bandwidth provided by the network. In operation, given that many users may be sharing the network bandwidth, the bandwidth allocated to any one particular user will be significantly less than
15 the theoretical maximum provided by the network. Thus, as files get larger, it is preferable for the workstations to be given direct access to a storage volume such that operational time is not lost while waiting for data to be transferred. For example, an A4 full colour image may consist of a total of 30 Mbytes of data. When transmitted over typical networks, a transfer
20 duration of several minutes may take place before the totality of the data has been received.

A problem with providing direct access to discs is that only one workstation may be given access to the data and in order for the data to be loaded into another machine, it may be necessary to physically move
25 transferrable discs, such as SCSI optical discs. Systems also exist under which a plurality of users may share direct access to a data storage device

and, consequently, measures must be implemented to remove the risk of contention problems. Thus, a particular workstation must release access to a particular file or disc partition before any of the other workstations may be allowed to write to that file.

5 In known systems, system specific software must be loaded into each workstation, so that each workstation is provided with instructions relating to the contention protocols. In addition, a plurality of workstations are given access to the shared volume by effectively dividing the volume into a plurality of partitions. Thus, in this way, a first workstation may write and
10 read data to a first partition of the disc, with a second workstation writing and reading to a second partition of the disc. At a later date, the first workstation may release the first partition, thereby allowing another workstation to be given access to this partition. In this way, a plurality of workstations may each access partitions within the volume without the data needing to be
15 transferred, thereby significantly improving operational performance.

 A problem with the above arrangement is that the partitioning of the disc may result in substantial storage regions being taken up that are only available for one workstation at any one time but do not actually contain valid data. Thus, for example, ten partitions of a very large disc volume may each
20 contain a relatively small amount of data. However, although a substantial amount of empty space remains on the disc, as far as the system is concerned, it would not be possible for this space to be allocated to another workstation, given that, as far as the system is concerned, the storage volume is fully allocated.

25 According to a first aspect of the present invention, there is provided a method of storing data wherein a plurality of user terminals access a large

storage volume, comprising steps of emulating the presence of a logical disc drive having a predetermined capacity; dividing said storage volume into a plurality of storage regions, wherein each of said regions is smaller than the size of an emulated logical disc drive; and mapping physical regions of data to an emulated drive dynamically as additional storage is required, up to said predetermined capacity.

Thus, in accordance with said first aspect, a workstation may be given access to a logical disc drive which it perceives as having a predetermined capacity. For example, the predetermined capacity may be similar to that provided by an optical disc providing 600 Mbytes of storage. However, physical storage locations on the large storage volume are only allocated, region by region, as the workstation demands additional storage through the writing of larger files to the disc.

In a preferred embodiment, a look-up table is associated with each accessible logical drive and a particular look-up table is loaded when its associated logical drive is selected.

According to a second aspect of the present invention, there is provided apparatus for storing data, having a plurality of user terminals and means for each of said terminals to be given access to said stored data, comprising means for emulating the presence of a logical disc drive having a predetermined capacity; means for dividing a storage volume into a plurality of storage regions, wherein each of said regions is smaller than the size of an emulated logical disc drive; and mapping means for mapping said physical regions of data to an emulated drive dynamically as additional storage is required, up to said predetermined capacity.

The system will now be described by way of example only, with reference to the accompanying Figures, in which:

Figure 1 shows an environment in which a plurality of workstations have access to a shared storage volume including a shared file server;

5 Figure 2 details the shared file server identified in Figure 1;

Figure 3 illustrates an application of the system shown in Figure 1; and

Figure 4 shows a schematic representation of the system, including the dynamic allocation of storage regions.

10 An environment in which a plurality of users have access to a shared storage volume is illustrated in Figure 1. In the environment shown in Figure 1, each workstation is provided with a processor 15, a visual display unit 16, an interface device in the form of a keyboard and/or a mouse or trackerball etc. 17 and a local disc drive storage device 18.

15 Each processor 15 is connected to a server interface 19 which allows said processors 15 to communicate with a shared file server 20. The file server 20 is connected to typically five physical hard disc drives 21, 22, 23, 24 and 25. This disc drive combination provides typically thirty-six Gbytes of storage with an access speed of typically 10 Mbytes per second.

20 Disc drives 21 to 25 may be configured as a redundant array, commonly referred to as a redundant array of inexpensive discs (RAID). In the preferred implementation, five discs are provided and the coding used to write data to the disc is commonly referred to as RAID 5. Thus, under this

protocol, redundant data is written to the discs such that if one of the drives becomes inoperable or suffers irretrievable damage, all of the data can be reconstituted from the remaining four drives.

5 Data is written to the drives in the form of identifiable blocks or regions of a predetermined length. The size of these blocks is determined from a trade-off between disc space optimisation and disc fragmentation. However, the system is primarily designed for storing large graphics files, therefore blocks may be quite large and it is proposed that said blocks should have a size between two Mbytes and thirty-two Mbytes. Similarly, it is
10 possible that the block size could be configurable for a particular application.

In operation, a user issues commands under software control which effectively result in a logical drive being made available by the server 20. Communication between the user and the server 20 is effected via the interface 19 and as far as the user is concerned, interface 19 presents a
15 standard small computer serial interface (SCSI) to the processor 15. Once a logical disc has been established, the user may access this drive.

The user's workstation receives data to the effect that it has been given access to a disc of a predetermined size, say 600 Mbytes for example, but in actuality, physical space is only allocated dynamically in regions as storage
20 space for the storage of actual data is required.

Thus, in the system shown in Figure 1 the server does not immediately allocate 600 Mbytes of storage to a user when access to a 600 Mbyte logical drive is requested. Space on drives 21 through 25 is not divided into 600 Mbytes (or similar) partitions. Drives 21 through 25 are divided into blocks

of between two and thirty-two Mbytes and blocks are only written to as data becomes available.

For the benefit of this illustration, it will be assumed that storage space on drives 21 through 25 has been divided into blocks of two Mbytes, thereby making two Mbyte blocks available for data storage purposes. As data is written to the drives, via an interface 19, said data will occupy one of said two Mbyte blocks. As the volume of data increases beyond two Mbytes, the server 20 will identify a new block of two Mbytes and data originating from a user will then continue to be written to this new two Mbyte block. Thus, for example, if a user has written a total of five Mbytes, the server is required to maintain a list of where these five Mbytes actually reside on the drives, in terms of three two-Mbyte blocks. However, as far as the user is concerned, five Mbytes of data have been written to on a logical drive having 600 Mbytes of available capacity.

Data is conventionally written to disc drives in terms of identifiable blocks. As far as the user is concerned, data is written to as blocks on a 600 Mbyte logical drive, which are in turn mapped onto real blocks on the RAID. However, the logical blocks may be written to in a substantially similar way to that in which real drives would be re-written to. Thus, it is not necessary for data to be written to the logical drives in what appears to be a contiguous region of disc space. Although the actual storage allocated for a logical drive is distributed over the RAID, the logical drives may appear, from the user's point of view, to be fragmented themselves. Thus, logical blocks of data may appear displaced over a logical drive, effectively emulating the presence of fragmentation on the logical disc. The system emulates such a situation by providing mapping firstly of blocks to logical drive locations and then mapping from logical drive locations to block locations on the RAID.

Many users may be given access to many virtual drives, allowing data to be accessed via many workstations without actually being transferred over a network. However, when capacity is allocated it is not wasted, in that blocks of two Mbytes are only allocated as actual storage is required.

5 In a preferred embodiment, it is envisaged that a server 20 would allow up to sixteen users to be connected thereto, although provision is made for server boxes to be connected in tandem, thereby providing access to a further 16 users for each box so connected.

10 The server 20 is detailed in Figure 2. Internally, a 32 bit parallel bus 25 provides communication between user interface circuits 26, disc drive interfaces 27, an internal processing unit 28 and internal program and data memory 29.

15 The server 20 is connected to each user interface 19 via a respective interface circuit 26 via two coaxial cables 30, providing a bi-directional link capable of conveying 100 Mbytes per second. Similarly, disc interface circuits 27 provide a parallel access to disc drives 21 through 25 and using connections of this type, it is necessary for disc drives 21 through 25 to be in close proximity to server box 20. In practice, the combination of server 20 along with disc drives 21 through 25 could be housed in a common housing with a shared power supply. However, coaxial cables 30 allow the users to be positioned at a significant distance from the server 20 and the interfaces are such that they will allow runs in excess of 100 metres. Thus, these serial connections are similar or may take advantage of high speed ethernet links.

20

In an alternative embodiment, user processors 15 are connected to the server 20 via conventional SCSI interfaces which, although reducing the overall complexity of the system, also reduce the maximum distance between the server 20 and the processors 15.

5 An application of the system is illustrated in Figure 3. At step 41 a user identifies a logical disc, either by running server related software or, alternatively, in response to manual operations of a device connected to interface 19. Thus, if it is not possible to embed server software within a user's terminal, it is possible to provide interfaces 19 with additional control
10 devices such that, in response to manual operation of switches etc., commands are sent to server 20 so as to establish a logical disc connection.

Communication of this type, allowing a user to send commands to the server 20, is achieved using vendor unique command blocks, which are data areas provided for specific proprietary applications within the SCSI standard.
15 Thus, in response to user originating commands, the server is instructed at step 42 to the effect that a user requires access to a logical drive.

For each logical drive which may be made available to the users, it being noted that once a logical drive has been established by any particular user, other users may be given access to it, it is necessary for the server 20
20 to create a sector mapping table for that particular logical drive. Thus, in response to commands generated by a user's processor, establishing logical sectors of a SCSI disc, it is necessary for the server 20 to map these logical sectors onto physical blocks or groups of physical blocks stored within the physical drives 21 through 25. At the CPU 28, reference is made to a look-
25 up table stored within memory 29 which, as previously stated, identifies physical data blocks held by the redundant disc array. Thus, the CPU is

required to generate the sector instructions relevant for the physical drives 21 through 25, which are issued to respective ones of said drives via respective interface circuits 27.

5 Once a user has requested use of a logical drive, the server identifies the space available to the user at step 44, in response to which the user may identify particular files to be written to or read from the logical drive.

10 At step 46 it is determined whether the user wishes to write data to or read data from a logical drive. If data is being written to the drive, an enquiry is made at step 47 as to whether space is available on the last block to be written to. If space is available, data is written to the next identified block at step 48. Alternatively, if sufficient space is not available on the last block, a new block is selected at step 49 and data is written to this block at step 50.

15 If a read operation is identified at step 46, the physical blocks to be read are identified at step 51, the data is read at step 52 and supplied to the requesting user in a suitable form. Thereafter, the process may be repeated and further identifications may be made at step 41.

20 A schematic representation of the system is illustrated in Figure 4. At a workstation, a user is presented with a user interface, capable of providing an environment for allowing existing logical drives to be selected and providing the capacity for new drives to be defined.

 The user interface 61 is in turn supported by a local operating system 62. Thus, an operator makes a file selection via user interface 61 and it is

then necessary for the local operating system 62 to generate commands which may be interpreted by the physical storage system.

As far as the local operating system 62 is concerned, the system is making access to conventional SCSI disc drives. Thus, the local operating system 62 communicates with a network interface, illustrated as 63 and physically consisting of interface 19 shown in Figure 1. The network interface 63 receives standard SCSI commands from the local operating system 62 and in turn generates modulated data for transmission over the serial link, shown as 64, connecting the network interface 63 to a server interface 64. A physical representation of server interface 64 is identified in Figure 2 as 26.

The transmission of data between the local operating system 62 and the network interface 63 conforms to establish SCSI protocols. However, the communication between network interface 63 and server interface 64 is internally defined by the system and is designed, in a preferred embodiment, to provide maximum data transfer rates over substantial lengths of cable, such as coaxial cable. Furthermore, the connection between the network interface 63 and the server interface 65 is bi-directional.

The network interface 63 is primarily concerned with driving signals generated by the local operating system 62 so that they may be transmitted over the serial communication link 64. However, the sector indications generated by the local operating system 62 are conveyed to the server interface 65 and it is the server operating system 66 which is required to convert SCSI sector selections into addresses for physical blocks located on the array of physical drives.

Thus, the server operating system 66 supplies addressing signals to the physical discs, identified as 67 whereafter data transfer is effected.

The server operating system 66 converts SCSI sector definitions into addressable physical data blocks by means of a look-up table, identified as 68.

5 A look-up table is defined for each logical drive and when a logical drive is selected by an operator its associated look-up table is loaded to an operating area of memory 29 within the server 20. Thus, within the operating system 66, a logical drive is identified, resulting in a table 68 being loaded. Thereafter, SCSI sector selections are supplied as inputs to said table, which

10 then results in addresses for physical data blocks being generated as outputs. Thus, as illustrated in Figure 4, the table 68 effectively points to addressable data blocks 69 in the array of physical data storing discs 21 through 25.

CLAIMS

1. A method of storing data wherein a plurality of user terminals access a large storage volume, comprising steps of
emulating the presence of a logical disc drive having a predetermined
5 capacity;
dividing said storage volume into a plurality of storage regions,
wherein each of said regions is smaller than the size of an emulated logical
disc drive; and
mapping said physical regions of data to an emulated drive
10 dynamically as additional storage is required, up to said predetermined
capacity.
2. A method according to claim 1, wherein a plurality of logical
drives are accessible to a user.
3. A method according to claim 2, wherein a look-up table is
15 associated with each accessible logical drive and a particular look-up table is
loaded when its associated logical drive is selected.
4. A method according to any of claims 1 to 3, wherein the logical
drives appear to a user system in a form compatible with a local physical disc
drive.
- 20 5. A method according to claim 4, wherein said logical drive is
connected via a small computer serial interface (SCSI).
6. A method according to any of claims 1 to 5, wherein the size
of said regions is variable and pre-set for a particular application.

7. Apparatus for storing data, having a plurality of user terminals and means for each of said terminals to be given access to said stored data, comprising

5 means for emulating the presence of a logical disc drive having a predetermined capacity;

means for dividing a storage volume into a plurality of storage regions, wherein each of said regions is smaller than the size of an emulated logical disc drive; and

10 mapping means for mapping said physical regions of data to an emulated drive dynamically as additional storage is required, up to said predetermined capacity.

8. Apparatus according to claim 7, including means for defining a plurality of logical drives, each accessible to a user.

15 9. Apparatus according to claim 8, including means for defining a look-up table associated with each of said logical drives and means for loading a particular look-up table when its associated logical drive is selected.

10. Apparatus according to any of claims 7 to 9, including means for presenting a logical drive to a system user in a form compatible with a local physical disc drive.

20 11. Apparatus according to claim 10, wherein said logical disc drive is connectable via a small computer serial interface (SCSI).

12. Apparatus according to any of claims 7 to 11, including means for pre-setting the size of said regions for a particular application.

13. Apparatus according to any of claims 7 to 11, wherein the size of said regions is variable in response to operator requests and said means for emulating the presence of the logical drive is arranged to supply data to a user terminal identifying the size of a logical drive being emulated.

5 14. A method of storing data substantially as herein described with reference to the accompanying Figures.

15. Apparatus for storing data substantially as herein described with reference to the accompanying Figures.



Application No: GB 9500173.1
Claims searched: 1-15

Examiner: Mr S J Probert
Date of search: 6 April 1995

Patents Act 1977
Search Report under Section 17

Databases searched:

UK Patent Office collections, including GB, EP, WO & US patent specifications, in:

UK CI (Ed.N): G4A AMX

Int CI (Ed.6): G06F 12/02

Other: Online Databases : WPI, INSPEC

Documents considered to be relevant:

Category	Identity of document and relevant passage	Relevant to claims
	None	

X	Document indicating lack of novelty or inventive step	A	Document indicating technological background and/or state of the art.
Y	Document indicating lack of inventive step if combined with one or more other documents of same category.	P	Document published on or after the declared priority date but before the filing date of this invention.
&	Member of the same patent family	E	Patent document published on or after, but with priority date earlier than, the filing date of this application.